Branch and Bound method for binary problems with the procedure that reduces dimension of problems

M. CHUDY
mchudy@wat.edu.pl

Military University of Technology, Faculty of Cybernetics
Institute of Computer and Information Systems
Kaliskiego Str. 2, 00-908 Warsaw, Poland

The relationships between elements a_{ij} of coefficient matrix, elements d_i of vector d and elements c_j of vector c in general binary problem are considered. Some of them allow us to establish the values of selected elements of feasible or optimal vector x. This procedure reduces the dimension of basic problem and can be install in branch and bound method. It gives positive effects.

Keywords: binary problem, branch and bound method, reducing dimension.

1. General remarks

Let us consider general binary problem in the following form: find

$$x^* \in S \subset B^n = \left\{ x \in E^n : x_j \in \{0, 1\},\ j \in N \right\}$$

such that

$$\sum_{j \in N} c_j x_j^* = \max_{x \in S} \sum_{j \in N} c_j x_j \tag{1}$$

where

$$S = \left\{ \begin{array}{l} x \in E^n : \sum a_{ij} x_j \le d_i \ i = \overline{1,m} \\ x_j \in \{0,1\},\ j \in N = \{1, 2, ..., n\} \end{array} \right\} \tag{2}$$

and $N = \{1, 2, ..., n\}$.

Without loss of generality we assume that

$$c_j \le 0 \quad j \in N \tag{3}$$

To illustrate presented in future properties we will analyze the following example of binary problem.

Example 1

$$\max -5x_1 - 6x_2 - 2x_3 - x_4$$

subject to

$$4x_1 - 2x_2 - x_3 - x_4 \le -1 = d_1$$
$$2x_1 + 5x_2 + 8x_3 - x_4 \le 6 = d_2$$
$$-x_1 + 8x_2 - x_3 + x_4 \le 5 = d_3$$
$$x_j \in \{0, 1\} \quad j = \overline{1, 4}$$

In this case one can easy indicates optimal solution. It is vector $x^* = (0, 0, 0, 1)$.

This simplicity comes from iterative applied property 1 of problem (1)–(2).

2. Selected properties of problem (1)

Property 1

If there exist i and j_* such that

$$\sum_{j \in J_i^-} a_{ij} + a_{ij_*} > d_i \tag{4}$$

where

$$J_i^- = \left\{ j : a_{ij} < 0 \right\}$$

then for each feasible solution x of problem (1)–(2) we obtain $x_{j_*} = 0$.

The proof results from the fact that for the value $x_{j_*} = 1$ constraint i-th does not hold with any values of variables x_j for $j \in N \setminus \{j_*\}$.

This property also holds if $J_i^- = \varnothing$ $\blacksquare$

Using property 1 for Example 1 in order given bellow we obtain:

for $i = 3$ and $j_* = 2$ Property 1 holds therefore $x_2 = 0$,

for $i = 2$ and $j_* = 3$ Property 1 holds, therefore $x_3 = 0$,

for $i = 1$ and $j_* = 1$ Property 1 holds, therefore $x_1 = 0$.

Additionally, to satisfy all the constrains we have to set $x_4 = 1$.

The generality of Property 1 is the following one.

Property 2

If for some i the following condition holds

$$\sum_{j\in J_i^-} a_{ij} + \sum_{j\in J_i^+} a_{ij} > d_i \qquad (5)$$

where

$$J_i^+ = \{ j : a_{ij} > 0 \}$$

then exists subset $J_i \neq \emptyset$, $J_i \subset J_i^+$ such that for each $j \in J_i$ we get $x_j = 0$ in every feasible solution of problem (1)–(2).

The proof results from the fact that if for each $j \in J_i^+$ $x_j = 1$, then for any values of variables x_j such that $j \in J_i^-$ the constraint i-th will not hold.

This property also holds if $J_i^- = \emptyset$ ∎

In some cases, for some elements of feasible solutions we have to set the value 1.

Property 3

If for some i such that $d_i < 0$ exists j_0 such that

$$\sum_{j\in J_i^-} a_{ij} \le d_i \quad \text{and} \quad \sum_{j\in J_i^- \setminus \{j_0\}} a_{ij} > d_i \qquad (6)$$

where

$$J_i^- = \{ j : a_{ij} < 0 \}, \quad j_0 \in J_i^-$$

then $x_{j_0} = 1$ in each feasible solution of problem (1)–(2). This property also is satisfied if $J_i^- \setminus \{ j_0 \} = \emptyset$.

The proof results from the fact that if variable x_{j_0} takes the value 0 then the constrain i-th will not be satisfied for any values of variables x_j such that $j \in N \setminus \{ j_0 \}$, $N = \{1, 2, ..., n\}$ ∎

Example 2

$$\max -6x_1 - 2x_2 - 5x_3 - 3x_4$$

subject to

$$x_1 - 3x_2 + x_3 - x_4 \le -2 = d_1$$
$$-x_1 + x_2 - 5x_3 + 2x_4 \le 3 = d_2$$
$$x_j \in \{0, 1\} \quad j = \overline{1, 4}$$

For $i = 1$, $j_0 = 2$ we have

$$a_{12} + a_{14} = -4 < -2 = d_1$$

and

$$a_{14} = -1 > -2 = d_1$$

therefore $x_2 = 1$.

Like Property 1 the Property 3 can be generalized.

Property 4

If for some i such that $d_i < 0$ holds than the following condition is satisfied

$$\sum_{j\in J_i^-} a_{ij} \le d_i \quad \text{and} \quad \sum_{j\in J_i^- \setminus J_i} a_{ij} > d_i \qquad (7)$$

where

$$J_i^- = \{ j : a_{ij} < 0 \}, \quad J_i \subset J_i^-, \quad J_i \neq \emptyset$$

Then for **some** $j \in J_i$ variables x_j are equal to $x_j = 1$ in any feasible solution of problem (1)–(2).

It also holds if $J_i^- \setminus J_i = \emptyset$.

The proof results from the fact that if **all** variables x_j $j \in J_i$ take the value 0, then the constrain i-th will not be satisfied for any values of variables x_j such that $j \in N \setminus J_i$ ∎

In certain cases one can easy indicates a feasible solution of problem (1)–(2).

Denote by a_j the j-th column of matrix A of problem (1)–(2).

Property 5

If there exists a set $J \neq \emptyset$, $J \subset N$ indexes of columns a_j of matrix A for which such condition holds

$$\sum_{j\in J} a_j \le d ,$$

then the vector x with elements $x_j = 1$ for $j \in J$ and $x_j = 0$ for $j \in N \setminus J$ is the feasible solution of problem (1)–(2).

The proof is evident ∎

The relationships between the columns of matrix A and elements of vector c create the optimal solutions.

Property 6

If there exists index $p \in N$ of column of matrix A such that

$$a_p \le d \ \text{ and } \ c_p = \max_{j\in N} c_j \qquad (8)$$

then the optimal solution of problem (1)–(2) has the form

$$x_p = 1, \quad x_j = 0 \quad j \in N \setminus \{ p \} \qquad (9)$$

To prove it we should observe that when condition (8) is satisfied the value of objective function decreases if we set $x_j = 1$ for any $j \neq p$. ∎

The Property 6 takes place in Example 2 for $p = 2$.

Property 7

If there exists a set $J \subset N$ of indexes of columns a_j of matrix A and index p such that

$$\sum_{j \in J} a_j < a_p \text{ and } \sum_{j \in J} c_j > c_p \qquad (10)$$

then $x_p = 0$ in optimal solution $x = x^*$ (if exists) of problem (1).

Proof

Let 2^N denotes the set of all subsets of set N.

For each $G \in 2^N$ we define formula $c(G) = \sum_{j \in G} c_j$

Denote by

$$D = \left\{ G \in 2^N : \sum_{j \in G} a_j \le d \right\} =$$

$$= \left\{ G \in 2^N : \sum_{j \in G} a_j x_j \le d, x_j = 1 \right\}$$

set of such subsets of indexes of columns (variables) which can take value 1 for feasible solution of problem (1)–(2).

Set G^* indexes of variables x_j, which are equal to 1 in optimal solution of problem (1) can be obtain by formula

$$c\left(G^*\right) = \max_{G \in \mathbf{D}} c\left(G\right) \qquad (11)$$

Index p which satisfies condition (10) can not belong to the set G^* because only two cases can take place:

1^0 $a_p \le d$, which causes $\sum_{j \in J} a_j < d$,

therefore with (10) the solution

$$x_j = 1 \text{ for } j \in J,\ x_j = 0 \text{ for } j \in N \setminus J \qquad (12)$$

is better than the solution

$$x_p = 1,\ x_j = 0 \text{ for } j \in N \setminus \{p\}, \qquad (13)$$

2^0 condition $a_p \le d$ **does not hold** but condition $\sum_{j \in J} a_j \le d$ is satisfied;

one can distinguish two cases:

– exists nonempty set $J_0 \subset N$ such that $a_p + \sum_{j \in J_0} a_j \le d$,

  therefore from the condition (10) with $c \le 0$ the condition $\sum_{j \in J} c_j > c_p + \sum_{j \in J_0} c_j$ is satisfied and it means that solution (12) is better than solution $x_p = 1$, $x_j = 1$ for $j \in J_0$ and $x_j = 0$ for other variables,

– does not exist nonempty set $J_0 \subset N$ such that $a_p + \sum_{j \in J_0} a_j \le d$

  therefore the solution containing $x_p = 1$ is not feasible. ■

To complement the properties mentioned above we will add one followed [2].

Property 8

If $d \ge 0$, then the feasible solution $x_j = 0$ for all $j \in N$ is optimal of problem (1)–(2).

The proof results for the assumption $c \le 0$ ■

The properties we described can be applied in branch and bound method to reduce the dimension of current considered binary problem on successive steps of this method.

It can decrease the time to obtain optimal solution.

3. Reducing procedure of number of variables in the problems defined on the subsets $S_k \subset S$

The binary problem on subset $S_k \subset S$, i.e. in the k-th vertex of the tree is defined in branch and bound method as follows:

find $x^* \in S_k \subset E^n$ such that

$$\sum_{j \in N} c_j x_j^* = \max_{x \in S_k} \sum_{j \in F_k} c_j x_j + \sum_{j \in N_k^+} c_j \cdot 1 + \sum_{j \in N_k^-} c_j \cdot 0 \qquad (14)$$

where

$$S_k = \left\{ \begin{array}{l} x \in E^n : \sum_{j \in F_k} a_{ij} x_j \le r_i^k,\ i = \overline{1, m}, \\ \quad x_j \in \{0,1\},\ j \in F_k, \\ \quad x_j = 1\ j \in N_k^+, x_j = 0, j \in N_k^- \end{array} \right\} \qquad (15)$$

$k = 0,1,...$

and

$N_k^+ = \{ j \in N : x_j = 1 \}$, $N_k^- = \{ j \in N : x_j = 0 \}$,

$N = \{1, 2, ..., n\}$, $F_k = N \setminus (N_k^+ \cup N_k^-)$ $S_k \subset S$,

where

$$c_j \le 0 \quad j \in F_k. \qquad (16)$$

We will name these variables x_j, $j \in F_k$ as the variables of non-fixed values.

CPR-Counting procedure for reducing dimension of problem (14)–(15).

1. Implement the data of problem (14)–(15). Set: $B_k^+ := \varnothing$, $B_k^- := \varnothing$.
   Comment: B_k^+, B_k^- are the supporting sets in computing.
   When we check the properties 1, 3, 6, 8 for $k \ge 0$ we consider vector r^k instead of vector d.
2. Check the condition $F_k = \varnothing$.
   If YES, go to 3.
   If NO, go to 4.
3. Create the vector $x(k)$ setting:
   $x_j := 1$ for $j \in N_k^+$ and $j \in B_k^+$,
   $x_j := 0$ for $j \in N_k^-$ and $j \in B_k^-$.
   Go to 9.
4. Check the Property 8 in the problem (14)–(15).
   If YES,
   set: $B_k^- := B_k^- \cup F_k$, $F_k := \varnothing$,
   go to 2.
   If NO, go to 5.
5. Check the Property 6 in the problem (14)–(15).
   If yes,
   set: $B_k^+ := B_k^+ \cup \{p\}$, $B_k^- := B_k^- \cup F_k \setminus \{p\}$, $F_k := \varnothing$,
   $x_p = 1$ in the problem (14), (15). Compute new values of the elements of vector r^k, go to 2.
   If NO, go to 6.
6. Check the Property 1 in the problem (14)–(15).
   If YES,
   set: $B_k^- := B_k^- \cup \{j_*\}$, $F_k := F_k \setminus \{j_*\}$, $x_{j_0} = 0$
   in the problem (14), (15), compute new values of elements of vector r^k, go to 2.
   If NO, go to 7.
7. Check the Property 3 in the problem (14)–(15).
   If YES,
   set: $B_k^+ := B_k^+ \cup \{j_0\}$, $B_k^- := F_k \setminus \{j_0\}$, $x_{j_0} = 1$
   in the problem (14)–(15), compute new values of elements of vector r^k, go to 2.
   If NO, go to 8.
8. Set: $F_k := F_k \setminus (B_k^+ \cup B_k^-)$. End.
   Comment: it is impossible to reduce more variables of non-fixed values.
9. Check if the vector $x(k) \in S$.
   If YES, go to 10.
   If NO, go to 11.
10. The solution $x^*(k) = x(k)$ is the optimal solution of problem (14)–(15). End.
    Comment: based on the considered properties all variables obtain the fixed values.
11. The problem (14)–(15) does not have any feasible solution. End.

Example 3

The following binary problem is given:

$$\max\ -5x_1 - 7x_2 - 10x_3 - 3x_4 - x_5 \qquad (17)$$

subject to

$$-x_1 + 3x_2 - 5x_3 - x_4 + 4x_5 \le -2 = d_1 = r_1^0$$
$$2x_1 - 6x_2 + 3x_3 + 2x_4 - 2x_5 \le 0 = d_2 = r_2^0$$
$$x_2 - 2x_3 + x_4 + x_5 \le -1 = d_3 = r_3^0$$
$$x_i \in \{0,1\} \quad i = \overline{1,5}$$

This problem is solved in [1] using branch and bound method. The optimal solution is $x^* = (0,1,1,0,0)$.

Let us apply the procedure CPR to the problem (17).

Properties 8, 6 and 1 do not work.

One can apply property 3 to the variable x_3, then for $i = 3$ $j_0 = 3$. Therefore $x_3 = 1$.

Problem (17) after applying the step above is:

$$\max\ -5x_1 - 7x_2 - 3x_4 - x_5 - 10$$

subject to

$$-x_1 + 3x_2 - x_4 + 4x_5 \le 3 = r_1^0$$
$$2x_1 - 6x_2 + 2x_4 - 2x_5 \le -3 = r_2^0$$
$$x_2 + x_4 + x_5 \le 1 = r_3^0$$
$$x_1, x_2, x_4, x_5 \in \{0,1\} \qquad (18)$$

Properties 8, 6 and 1 do not work in problem (18).

One can apply Property 3 to the variable x_2, then $i = 2$ $j_0 = 2$ and $x_2 = 1$.

The problem (18) after applying the step above is:

$$\max\ -5x_1 - 3x_4 - x_5 - 17 \qquad (19)$$

subject to

$$-x_1 - x_4 + 4x_5 \le 0 = r_1^0$$
$$2x_1 + 2x_4 - 2x_5 \le 3 = r_2^0$$
$$x_4 + x_5 \le 0 = r_3^0$$

$$x_1, x_4, x_5 \in \{0,1\}$$

The Property 8 works in the problem (19). Therefore $x_1 = 0,\ x_4 = 0,\ x_5 = 0$. This is the optimal solution of problem (19). Matching the results of reducing the problem (17) with $x_2 = 1, x_3 = 1$ we obtain optimal solution $x^* = (0,1,1,0,0)$. In this case the procedure CPR is very effective.

We apply the procedure CPR to known and very simple algorithm "indirect search" described in [2] and [1].

4. Procedure of modified indirect search algorithm

Assume that the relaxation of problem (14)–(15) has the following form

find $x^0(k) \in T_k \supset S_k$ such that

$$\sum_{j\in N} c_j x_j^0 = \max_{x\in T_k} \sum_{j\in F_k} c_j x_j + \sum_{j\in N_k^+} c_j \cdot 1 + \sum_{j\in N_k^-} c_j \cdot 0 \tag{20}$$

where

$$T_k = \left\{ \begin{array}{l} x : x_j \in \{0,1\} \quad j \in F_k , \\ x_j = 1 \ j \in N_k^+ ,\, x_j = 0 \ \ j \in N_k^- \end{array} \right\} \tag{21}$$

To indicate the variable x_p for dividing the vertex S_k we apply the following formula.

For (15) we describe the set:

$Q_k = \{\, i : d_i < 0\},\quad R_k = \{ j \in F_k : a_{ij} < 0,\ \ i \in Q_k \}$

For $k > 1$ we use the vector r^k instead of vector d.

For each $j \in R_k$ we compute

$$V_j = \max_{i=\overline{1,m}} \left(\max\{ 0, -r_i + a_{ij} \} \right)$$

Index p of variable x_p to divide the vertex (set) S_k into two subsets we compute from the formula

$$V_p = \min_{j\in R_k} V_j \tag{22}$$

Computing procedure for binary problems.

1. Implement the data of problem (1)–(2). Set: $k := 0$, $F_k := N$, $S_k = S$, $\overline{Z}_k = +\infty$, $\underline{Z}_k = -\infty$.
2. On the vertex S_k we make the following operations:

2.1. Create the sets.

2.2. W_k, F_k, N_k^+, N_k^-, Q_k.

2.3. Compute the assessment $\overline{Z}_k = \sum_{j\in N_k^+} c_j$ by solving relaxation (20)–(21).

2.4. Consider Q_k.

2.3.1. If $Q_k = \varnothing$ and exists such i, that $t_i > r_i$ or $\overline{Z}_k \le \underline{Z}_0$, then the vertex S_k one closes and goes to 3.
Comment: the formula $x^0(k) \in T_k \supset S_k$ means that i-th constraint cannot be fulfilled.

2.3.2. If $Q_k \ne \varnothing$ and $\overline{Z}_k > \underline{Z}_0$ or $Q_k = \varnothing$ then we apply procedure CPR for the problem (14)–(15) in the vertex S_k.

2.3.2.1. If the procedure CPR terminates in point 10 then we close the vertex S_k.
Count $\overline{Z}_k = \underline{Z}_k = \sum_{j\in N_k^+} c_j + \sum_{j\in F_k^+} c_j$.
Set $\underline{Z}_0 := \max\{\underline{Z}_0, \underline{Z}_k\}$.
Comment: one remembers vector x that gives the assessment $\underline{Z}_0 := \max\{\underline{Z}_0, \underline{Z}_k\}$.
Compute and remember the vector $x(k)$ with elements $x_j = 1$ for $N_k^+ \cup F_k^+$, and $x_j = 0$ for $j \in N_k^- \cup F_k^-$. Go to 3.

2.3.2.2. If the procedure CPR terminates in point 11 then we close the vertex S_k and go to 3.

2.3.2.3. If the procedure CPR terminates in point 8, then we set successively:
$N_k^+ := N_k^+ \cup B_k^+$,
$N_k^- := N_k^- \cup B_k^-$, $\ F_k := F_k \setminus \left(B_k^+ \cup B_k^-\right)$
$W_k := N \setminus \left(N_k^+ \cup N_k^- \cup F_k\right)$.
Count $\overline{Z}_k = \sum_{j\in N_k^+} c_j$. Go to 4.

3. If the set of active vertex is empty go to 5, otherwise take the active vertex according established rule and go to 2.
4. Divide the set S_k into two subsets taking the variable x_p from the formula (22) after computed the set R_k.
Compute the successors of vertex S_k and set the labels of them according the rule of labelling the vertexes of the tree.
$S_{k_1} = S_k \cap \{ x : x_p = 1\}$,
$S_{k_2} = S_k \cap \{ x : x_p = 0\}$
Set $k := k_1$ i.e.. the index of this successor of vertex S_k for which $x_p = 1$. Go to 2.

5. If $\underline{Z}_0 = -\infty$, then the problem (1)–(2) does not have any feasible solution i.e. $S_0 = \varnothing$
If $\underline{Z}_0 > -\infty$, then the solution x which gives $\underline{Z}_0$ is the optimal solution of problem (1), i.e. $x = x^*$. End.

5. Conclusions

General binary problem belongs to the class of NP-hard ones.

The complexity of procedures that verifies the properties 1, 3, 6, 8 in CPR equals at most $O(n^2)$.

Applying CPR procedure we improve the complexity of current considered binary problem remaining inside the NP-hard class. In general such operation takes advantage of computing.

We know that there exists such binary problems which dimension cannot be reduced using considered properties. Therefore we expect some advantages only in such cases, when the CPR can be effectively used.

6. Bibliography

[1] Chudy M., *Wybrane algorytmy optymalizacji*, EXIT, Warszawa, 2014.

[2] Garfinkel R.S., Nemhauser G.L. *Programowanie całkowitoliczbowe*, PWN, Warszawa, 1978.

[3] Hromkovic J., *Algorithmics for Hard Problems. Introduction to Combinatorial Optimization, Randomization, Approximation and Heuristics*, Springer-Verlag, Berlin, Heidelberg, 2001.

[4] Martello S., Toth P., *An exact algorithm for the two-constraint 0-1 knapsack problem, Operations Research,* No. 5, 826–835 (2003).

[5] Nemhauser G.L., Wolsey L.A., *Integer and Combinatorial Optimization*, John Wiley & Sons, 1999.

[6] Sierksma G., *Linear and Integer Programming. Theory and Practice*, Marcel Dekker, Inc. New York, 1996.

Metoda podziału i oszacowań dla zadań binarnych z procedurą, która redukuje wymiar zadań

M. CHUDY

Przedstawiono kilka własności problemów binarnych, które pozwalają redukować wymiar zadania poprzez wyszukiwanie i ustalanie wartości niektórych zmiennych. W dopuszczalnych wektorach binarnych danego zadania wartości te muszą być przyjęte. Procedurę wykorzystującą te własności wmontowano w metodę podziału i oszacowań, w szczególności do algorytmu przeglądu pośredniego dla zadań binarnych.

Słowa kluczowe: problem binarny, metoda podziału i oszacowań, redukcja wymiaru.